BERT

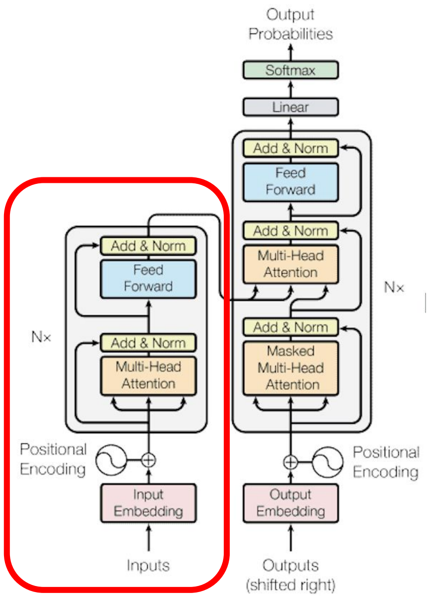
Pre-training of Deep Bidirectional Transformers For Language Understanding

Hunner Markus	01503441
Meier Ronja	12433721
Steinegger Benno	12117772

BERT[1]

Bidirectional **E**ncoder **R**epresentations from **T**ransformers

» Encoder-Only model

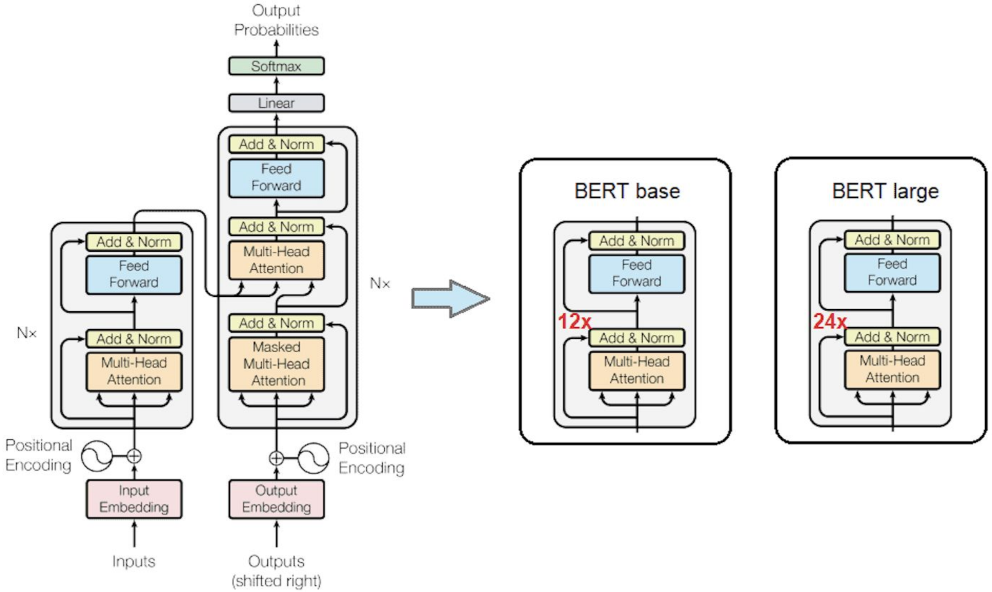


[1] Devlin, Jacob, et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding" 2019.

BERT[1]

Bidirectional **E**ncoder **R**epresentations from **T**ransformers

» Encoder-Only model



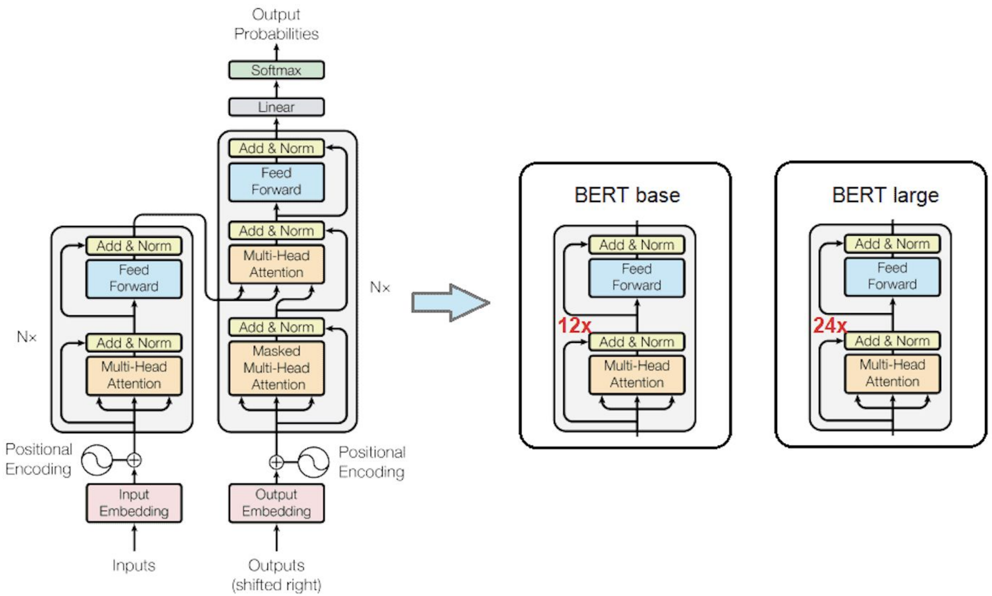
[1] Devlin, Jacob, et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding" 2019.

BERT[1]

Bidirectional **E**ncoder **R**epresentations from **T**ransformers

» Encoder-Only model using bidirectional attention

- Deep understanding of input text
- Excels at NLP tasks like text classification or question answering



[1] Devlin, Jacob, et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding" 2019.

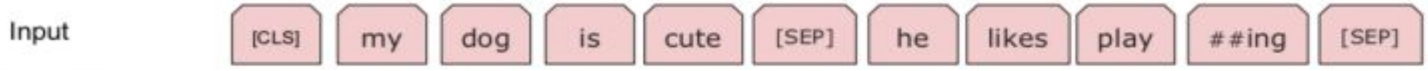
BERT[1]

» WordPiece Tokenization

- "unhappiness" → ["un", "##happiness"], "playing" → ["play", "##ing"]
- Small vocabulary → Efficient mapping of text to embeddings & covering of infrequent terms

» Special tokens

- [CLS] Classification token, used as pooling operator to get a single vector per sequence
- [SEP] Used to indicate a second sentence
- [MASK] Used in the masked language model, to predict this word
- [PAD] Added to sequences to ensure all inputs in a batch have the same length



[1] Devlin, Jacob, et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding" 2019.

BERT - Model

» Model architecture:

- n layers of stacked transformers
- Every transformer's layer receives as input the output of the previous layer
- Model head depending on problem

» Pre-training & workflow

- **Pre-training** a large model on huge datasets needs a lot of computation
- Download a pre-trained model, attach a head fit for the problem task, and **fine-tune** on own dataset

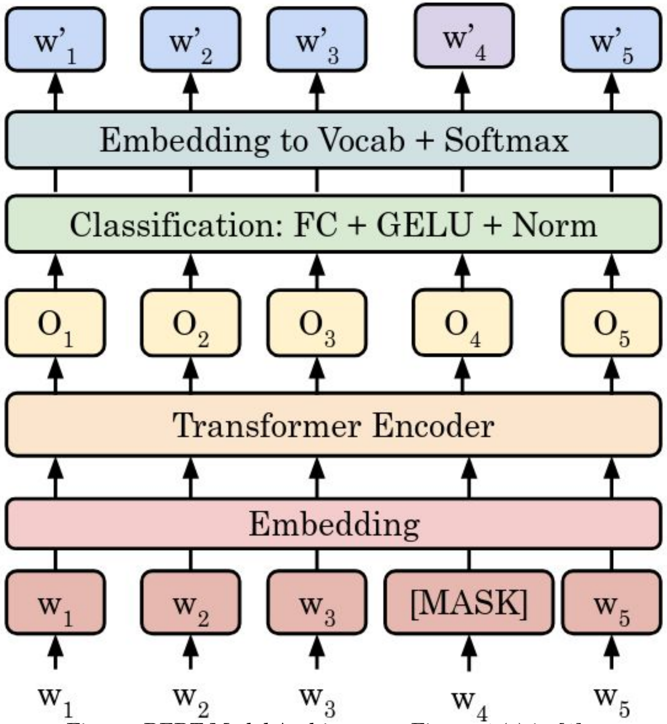


Figure: BERT Model Architecture. Figure 4 (a) in [1]

[1] Chitty-Venkata, et al. "A Survey of Techniques for Optimizing Transformer Inference" 2023.

Pre-Training - Motivation

- For many tasks we do not have much training data
- Large models need a lot of data to work well

Solution: Train unsupervised on task-agnostic data

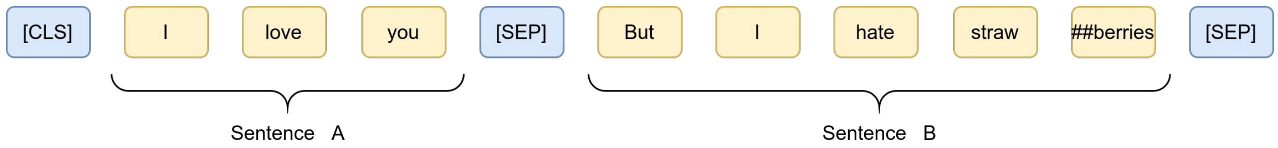
  - Train model about meaning of words and patterns in language
  - Train without labels, and instead predict word and sentence positions

Pre-Training

» Take any large text corpus …

» … and form sentence pairs

- 50% of the time B is the actual next sentence that follows A (labeled as *IsNext*), and 50% of the time it is a random sentence from the corpus (labeled as *NotNext*).



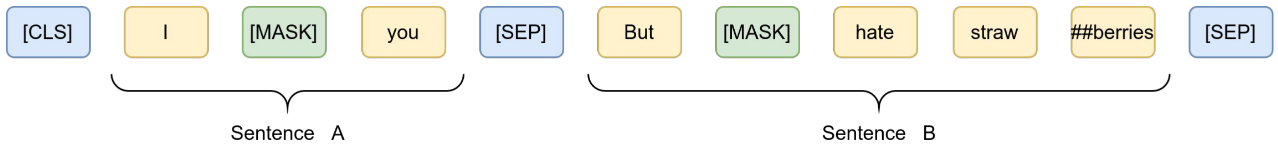
Pre-Training

» Take any large text corpus …

» … and form sentence pairs

- 50% of the time B is the actual next sentence that follows A (labeled as *IsNext*), and 50% of the time it is a random sentence from the corpus (labeled as *NotNext*).

» … and replace 15% of all tokens by [MASK]

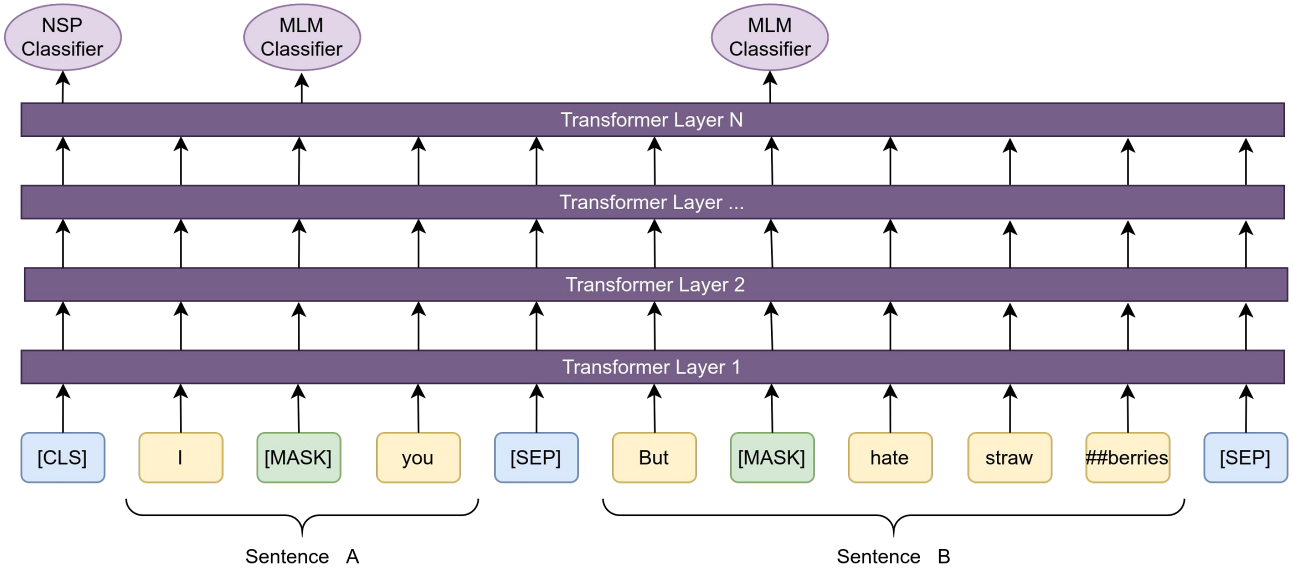


Pre-Training

» … send it through the model and

» … give **[CLS]** token result to Next-Sentence-Prediction (NSP) classifier

» … give **[MASK]** token results to Masked-Language-Modeling (MLM) classifier



Pre-Training

- … send it through the model and
- … all classifier contribute to the loss. → They do not learn separately!

- Someone with lots of compute and time **pre-trains** a large model …
- … and we download it and **fine-tune** on our own data.

Fine-Tuning

» To fine-tune BERT we simply replace the model heads with task-specific ones.

» For example: Add a Sequence Classification onto the **[CLS]** token output.

- We do not need a pooling layer, as we trained the model to encode all necessary information in the [CLS] vector output.

» Fine-tuning for 2 to 4 epochs is usually sufficient.

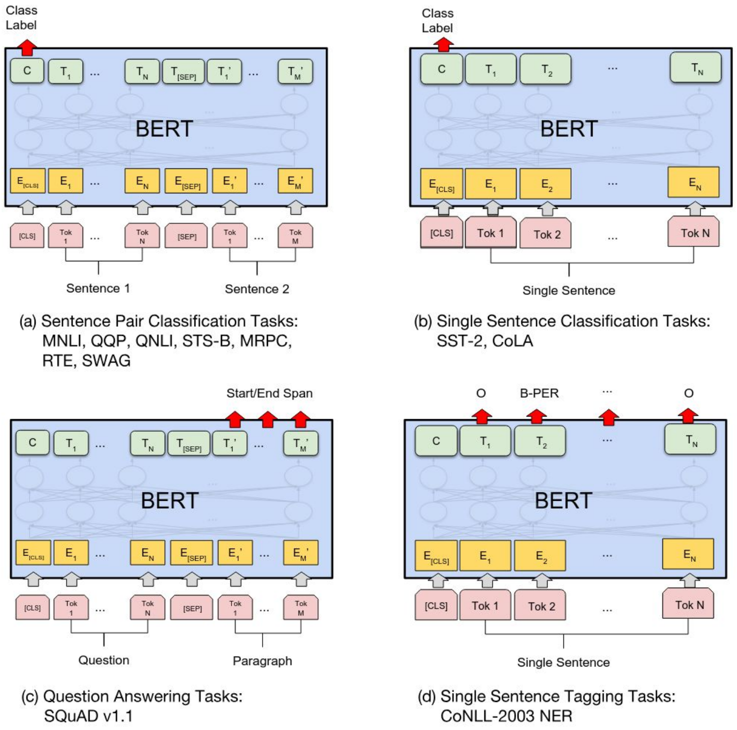


Figure: Illustrations of Fine-tuning BERT on Different Tasks.
Figure 4 in [1]

[1] Devlin, Jacob, et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding" 2019.

ModernBERT

Important architectural changes

ModernBERT[3] - Structural Changes

» Bias Terms

- Disable bias in all linear layers expect the in final decoder linear layer
- Disable in Layer Norms

» Positional Embeddings

- Rotary positional Embeddings (RoPE) instead of absolute positional embeddings

» Activation

- GeGLU based on original BERT's GeLU
- Gated-Linear Units based

» Normalization

- Pre-normalization block & standard layer normalization to stabilize training

[3] B. Warner et al., "Smarter, Better, Faster, Longer: A Modern Bidirectional Encoder for Fast, Memory Efficient, and Long Context Finetuning and Inference," 2024, arXiv. doi: 10.48550/ARXIV.2412.13663.

ModernBERT - Model Design

» Remove NSP

» Masked Language Modeling (30% instead of 15%; from MosaicBERT)

» Tokenizer

  - BPE tokenizer based on OLMo
  - Same special tokens (e.g. [CLS], [SEP], …)

» Deeper & Narrow Layer

  - 22 (base), 28 (large) layers

» Higher Context Length of 8192

» Learning Rate & Batch Size Scheduler

ModernBERT - Efficiency Improvements

» Attention

  - Alternating Attention: alternates between global and a sliding window attention
  - Flash Attention 3

» Unpadding

  - Remove padding tokens

» Use torch.compile

GLUE Fine-tuning

General **L**anguage **U**nderstanding **E**valuation

GLUE[2]

General **L**anguage **U**nderstanding **E**valuation

» A whole suite of **9 sentence classification problems** in English.

- 6 classification problems with 2 labels (**CoLA**, **MRPC**, **QNLI**, **QQP**, **RTE**, **SST2**)
- 2 classification problems with 3 labels (**MNLI-m**, **MNLI-mm**)
- 1 regression problem within values of 1 to 5 (**STSB**)

» A set of standardized public datasets.

- 8 training datasets (Problems **MNLI-m**, **MNLI-mm** share a train set)
- 9 validation datasets
- 9 tests sets with *censored labels*

[2] Wang, Alex, et al. "GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding." 2018.

GLUE[2]

General Language Understanding Evaluation

» A whole suite of **9 sentence classification problems** in English.

- 6 classification problems with 2 labels (**CoLA**, **MRPC**, **QNLI**, **QQP**, **RTE**, **SST2**)
- 2 classification problems with 3 labels (**MNLI-m**, **MNLI-mm**)
- 1 regression problem within values of 1 to 5 (**STSB**)

» A set of standardized public datasets.

- 8 training datasets (Problems **MNLI-m**, **MNLI-mm** share a train set)
- 9 validation datasets
- ~~9 tests sets with *censored labels*~~

⚠️ We need a labeled test set to provide results …

Solution: Extract a holdout from labeled training or validation set to serve as new test set

[2] Wan

GLUE - Results of original Paper

System	MNLI-(m/mm)	QQP	QNLI	SST-2	CoLA	STS-B	MRPC	RTE	**Average**
	392k	363k	108k	67k	8.5k	5.7k	3.5k	2.5k	-
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1
$\text{BERT}_{\text{BASE}}$	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
$\text{BERT}_{\text{LARGE}}$	**86.7/85.9**	**72.1**	**92.7**	**94.9**	**60.5**	**86.5**	**89.3**	**70.1**	**82.1**

"Table 1: GLUE Test results, scored by the evaluation server (https://gluebenchmark.com/leaderboard). The number below each task denotes the number of training examples. The "Average" column is slightly different than the official GLUE score, since we exclude the problematic WNLI set. BERT and OpenAI GPT are single-model, single task. F1 scores are reported for QQP and MRPC, Spearman correlations are reported for STS-B, and accuracy scores are reported for the other tasks." [1]

[1] Devlin, Jacob, et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding" 2019.

GLUE - Our Results

System	MNLI (m/mm) 392k	QQP 363k	QNLI 108k	SST-2 67k	CoLA 8.5k	STS-B 5.7k	MRPC 3.5k	RTE 2.5k	**Avg.** -
Current Top 3 Models according to GLUE leaderboard:									
Turing ULR v6	92.5/92.1	76.4	96.7	97.5	73.3	93.1	94.2	93.6	89.9
Vega v1	92.1/91.9	76.7	96.7	**97.9**	**73.8**	93.1	**94.5**	92.4	89.9
Turing NLR v5	**92.6/92.4**	76.4	**97.9**	97.6	72.6	**93.3**	93.8	**94.1**	**90.0**
Models compared to in original paper:									
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1
Results of original BERT paper:									
$\text{BERT}_{\text{BASE}}$	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
$\text{BERT}_{\text{LARGE}}$	86.7/85.9	72.1	92.7	94.9	60.5	86.5	89.3	70.1	82.1
Our reproduction:									
$\text{BERT}_{\text{BASE}}$	86.4/84.2	87.7	89.4	94.6	54.8	84.9	85.6	64.7	81.4
$\text{BERT}_{\text{LARGE}}$	88.2/85.3	88.2	90.5	94.7	58.8	87.6	88.1	67.5	83.2
Our comparison with ModernBERT:									
$\text{ModernBERT}_{\text{BASE}}$	88.4/86.9	88.3	90.6	94.1	58.9	89.0	89.7	76.3	84.7
$\text{ModernBERT}_{\text{LARGE}}$	89.2/87.6	**88.7**	91.5	94.2	60.2	90.6	90.5	83.1	86.2

GLUE - Our Results

System	MNLI (m/mm) 392k	QQP 363k	QNLI 108k	SST-2 67k	CoLA 8.5k	STS-B 5.7k	MRPC 3.5k	RTE 2.5k	**Avg.** -
[illegible]									
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]									
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
Results of original BERT paper:									
$\text{BERT}_{\text{BASE}}$	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
$\text{BERT}_{\text{LARGE}}$	86.7/85.9	72.1	92.7	94.9	60.5	86.5	89.3	70.1	82.1
Our reproduction:									
$\text{BERT}_{\text{BASE}}$	86.4/84.2	87.7	89.4	94.6	54.8	84.9	85.6	64.7	81.4
$\text{BERT}_{\text{LARGE}}$	88.2/85.3	88.2	90.5	94.7	58.8	87.6	88.1	67.5	83.2
[illegible]									
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

GLUE - Our Results

System	MNLI (m/mm) 392k	QQP 363k	QNLI 108k	SST-2 67k	CoLA 8.5k	STS-B 5.7k	MRPC 3.5k	RTE 2.5k	**Avg.** -
Results of original BERT paper:									
$\text{BERT}_{\text{BASE}}$	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
$\text{BERT}_{\text{LARGE}}$	86.7/85.9	72.1	92.7	94.9	60.5	86.5	89.3	70.1	82.1
Our reproduction:									
$\text{BERT}_{\text{BASE}}$	86.4/84.2	87.7	89.4	94.6	54.8	84.9	85.6	64.7	81.4
$\text{BERT}_{\text{LARGE}}$	88.2/85.3	88.2	90.5	94.7	58.8	87.6	88.1	67.5	83.2

Overall, we achieve comparable prediction performance.

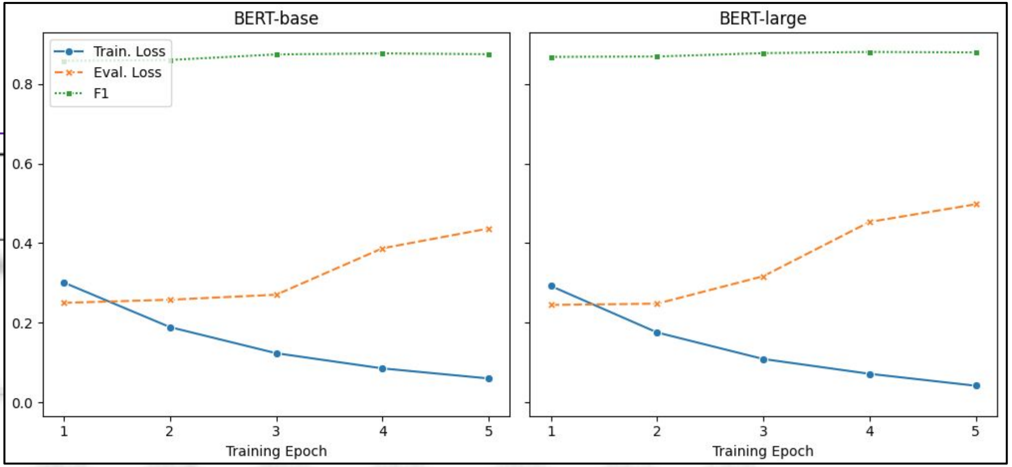
⚠️ Not a valid comparison: We do not evaluate on the same test dataset!

GLUE - Our Results

QQP dataset size

Original Train: 363k
Original Test: 390k
Our Train: 291k
Our Test: 72k

System		QQP 363k							
Results of original BERT paper:									
$\text{BERT}_{\text{BASE}}$	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
$\text{BERT}_{\text{LARGE}}$	86.7/85.9	72.1	92.7	94.9	60.5	86.5	89.3	70.1	82.1
Our reproduction:									
$\text{BERT}_{\text{BASE}}$	86.4/84.2	87.7	89.4	94.6	54.8	84.9	85.6	64.7	81.4
$\text{BERT}_{\text{LARGE}}$	88.2/85.3	88.2	90.5	94.7	58.8	87.6	88.1	67.5	83.2



⚠️ Not a valid comparison: We do not evaluate on the same test dataset!

GLUE - Our Results

System	MNLI (m/mm) 392k	QQP 363k	QNLI 108k	SST-2 67k	CoLA 8.5k	STS-B 5.7k	MRPC 3.5k	RTE 2.5k	**Avg.** -
[illegible]									
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]									
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
Results of original BERT paper:									
$\mathrm{BERT}_{\mathrm{BASE}}$	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
$\mathrm{BERT}_{\mathrm{LARGE}}$	86.7/85.9	72.1	92.7	94.9	60.5	86.5	89.3	70.1	82.1
Our reproduction:									
$\mathrm{BERT}_{\mathrm{BASE}}$	86.4/84.2	87.7	89.4	94.6	54.8	84.9	85.6	64.7	81.4
$\mathrm{BERT}_{\mathrm{LARGE}}$	88.2/85.3	88.2	90.5	94.7	58.8	87.6	88.1	67.5	83.2
Our comparison with ModernBERT:									
$\mathrm{ModernBERT}_{\mathrm{BASE}}$	88.4/86.9	88.3	90.6	94.1	58.9	89.0	89.7	76.3	84.7
$\mathrm{ModernBERT}_{\mathrm{LARGE}}$	89.2/87.6	**88.7**	91.5	94.2	60.2	90.6	90.5	83.1	86.2

GLUE - Our Results

🎉 ModernBERT$_{\text{BASE}}$ achieves prediction performance comparable to BERT$_{\text{LARGE}}$

System	MNLI (m/mm) 392k	QQP 363k	QNLI 108k	SST-2 67k	CoLA 8.5k	STS-B 5.7k	MRPC 3.5k	RTE 2.5k	**Avg.** -
Results of original BERT paper:									
BERT$_{\text{BASE}}$	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
BERT$_{\text{LARGE}}$	86.7/85.9	72.1	92.7	94.9	60.5	86.5	89.3	70.1	82.1
Our reproduction:									
BERT$_{\text{BASE}}$	86.4/84.2	87.7	89.4	94.6	54.8	84.9	85.6	64.7	81.4
BERT$_{\text{LARGE}}$	88.2/85.3	88.2	90.5	94.7	58.8	87.6	88.1	67.5	83.2
Our comparison with ModernBERT:									
ModernBERT$_{\text{BASE}}$	88.4/86.9	88.3	90.6	94.1	58.9	89.0	89.7	76.3	84.7
ModernBERT$_{\text{LARGE}}$	89.2/87.6	**88.7**	91.5	94.2	60.2	90.6	90.5	83.1	86.2

GLUE - Our Results

System	MNLI (m/mm) 392k	QQP 363k	QNLI 108k	SST-2 67k	CoLA 8.5k	STS-B 5.7k	MRPC 3.5k	RTE 2.5k	**Avg.** -
Our reproduction:									
$\text{BERT}_{\text{LARGE}}$	88.2/85.3	88.2	90.5	94.7	58.8	87.6	88.1	67.5	83.2
Our comparison with ModernBERT:									
$\text{ModernBERT}_{\text{BASE}}$	88.4/86.9	88.3	90.6	94.1	58.9	89.0	89.7	76.3	84.7
Speedup in training time $\text{BERT}_{\text{BASE}}$ / $\text{ModernBERT}_{\text{BASE}}$	1.38	1.24	1.38	0.99	0.76	1.15	1.18	1.36	1.18

🎉 $\text{ModernBERT}_{\text{BASE}}$ achieves prediction performance comparable to $\text{BERT}_{\text{LARGE}}$

🏁 $\text{ModernBERT}_{\text{BASE}}$ achieves significantly better runtime performance than $\text{BERT}_{\text{BASE}}$

GLUE - Our Results

System	MNLI (m/mm) 392k	QQP 363k	QNLI 108k	SST-2 67k	CoLA 8.5k	STS-B 5.7k	MRPC 3.5k	RTE 2.5k	**Avg.** -
Current Top 3 Models according to GLUE leaderboard:									
Turing ULR v6	92.5/92.1	76.4	96.7	97.5	73.3	93.1	94.2	93.6	89.9
Vega v1	92.1/91.9	76.7	96.7	**97.9**	**73.8**	93.1	**94.5**	92.4	89.9
Turing NLR v5	**92.6/92.4**	76.4	**97.9**	97.6	72.6	**93.3**	93.8	**94.1**	**90.0**
Models compared to in original paper:									
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1
Results of original BERT paper:									
$BERT_{BASE}$	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
$BERT_{LARGE}$	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
Our reproduction:									
$BERT_{BASE}$	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
$BERT_{LARGE}$	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
Our comparison with ModernBERT:									
$ModernBERT_{BASE}$	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
$ModernBERT_{LARGE}$	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

GLUE - Our Results

System	MNLI (m/mm) 392k	QQP 363k	QNLI 108k	SST-2 67k	CoLA 8.5k	STS-B 5.7k	MRPC 3.5k	RTE 2.5k	**Avg.** -
Current Top 3 Models according to GLUE leaderboard:									
Turing ULR v6	92.5/92.1	76.4	96.7	97.5	73.3	93.1	94.2	93.6	89.9
Vega v1	92.1/91.9	76.7	96.7	**97.9**	**73.8**	93.1	**94.5**	92.4	89.9
Turing NLR v5	**92.6/92.4**	76.4	**97.9**	97.6	72.6	**93.3**	93.8	**94.1**	**90.0**
Models compared to in original paper:									
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1

🤔 Many of the task in the GLUE benchmark can not be considered particularly "hard" problems anymore.

SQuAD Fine-tuning

Stanford Question Answering Dataset

SQuAD dataset

- Reading comprehension dataset
- Questions posed by crowdworkers on a set of Wikipedia articles
- Training dataset: 87599 | Validation dataset: 10570
- Focus on SQuAD 1.1 task
  - Extract the correct answer from a context to a given question
- SQuAD 2.0
  - Extends 1.1 by considering the possibility that no short answer exists in the provided paragraph → more realistic

SQuAD 1.1 results of the BERT paper

System	Dev EM	Dev F1	Test EM	Test F1
Top Leaderboard Systems (Dec 10th, 2018)				
Human	-	-	82.3	91.2
#1 Ensemble - nlnet	-	-	86.0	91.7
#2 Ensemble - QANet	-	-	84.5	90.5
Published				
BiDAF+ELMo (Single)	-	85.6	-	85.8
R.M. Reader (Ensemble)	81.2	87.9	82.3	88.5
Ours				
$\text{BERT}_{\text{BASE}}$ (Single)	80.8	88.5	-	-
$\text{BERT}_{\text{LARGE}}$ (Single)	84.1	90.9	-	-
$\text{BERT}_{\text{LARGE}}$ (Ensemble)	85.8	91.8	-	-
$\text{BERT}_{\text{LARGE}}$ (Sgl.+TriviaQA)	**84.2**	**91.1**	**85.1**	**91.8**
$\text{BERT}_{\text{LARGE}}$ (Ens.+TriviaQA)	**86.2**	**92.2**	**87.4**	**93.2**

SQuAD 1.1 results of the BERT paper

System	Dev EM	Dev F1	Test EM	Test F1
Top Leaderboard Systems (Dec 10th, 2018)				
Human	-	-	82.3	91.2
#1 Ensemble - nlnet	-	-	86.0	91.7
#2 Ensemble - QANet	-	-	84.5	90.5
Published				
BiDAF+ELMo (Single)	-	85.6	-	85.8
R.M. Reader (Ensemble)	81.2	87.9	82.3	88.5
Ours				
$BERT_{BASE}$ (Single)	80.8	88.5	-	-
$BERT_{LARGE}$ (Single)	84.1	90.9	-	-
$BERT_{LARGE}$ (Ensemble)	85.8	91.8	-	-
$BERT_{LARGE}$ (Sgl.+TriviaQA)	**84.2**	**91.1**	**85.1**	[illegible]
$BERT_{LARGE}$ (Ens.+TriviaQA)	**86.2**	**92.2**	**87.4**	**93.2**

Paper does not specify which 7 pre-training checkpoints and which fine-tuning seeds are used

TriviaQA problem with Training loss

SQuAD 1.1 results of the BERT paper

System	Dev		Test	
	EM	F1	EM	F1
Top Leaderboard Systems (Dec 10th, 2018)				
Human	-	-	82.3	91.2
#1 Ensemble - nlnet	-	-	86.0	91.7
#2 Ensemble - QANet	-	-	84.5	90.5
Published				
BiDAF+ELMo (Single)	-	85.6	-	85.8
R.M. Reader (Ensemble)	81.2	87.9	82.3	88.5
Ours				
$\text{BERT}_{\text{BASE}}$ (Single)	80.8	88.5	-	-
$\text{BERT}_{\text{LARGE}}$ (Single)	84.1	90.9	-	-
~~$\text{BERT}_{\text{LARGE}}$ (Ensemble)~~	~~85.8~~	~~91.8~~		
~~$\text{BERT}_{\text{LARGE}}$ (Sgl.+TriviaQA)~~	~~**84.2**~~	~~**91.1**~~	~~**85.1**~~	~~**91.8**~~
~~$\text{BERT}_{\text{LARGE}}$ (Ens.+TriviaQA)~~	~~**86.2**~~	~~**92.2**~~	~~**87.4**~~	~~**93.2**~~

SQuAD 1.1 our results

Model	EM		F1	
	Paper	Our	Paper	Our
BERT-Base	80.8	79.9	88.5	87.7
BERT-Large	84.1	83.3	90.9	90.5

SWAG Fine-tuning

A Large-Scale Adversarial Dataset for Grounded Commonsense Inference

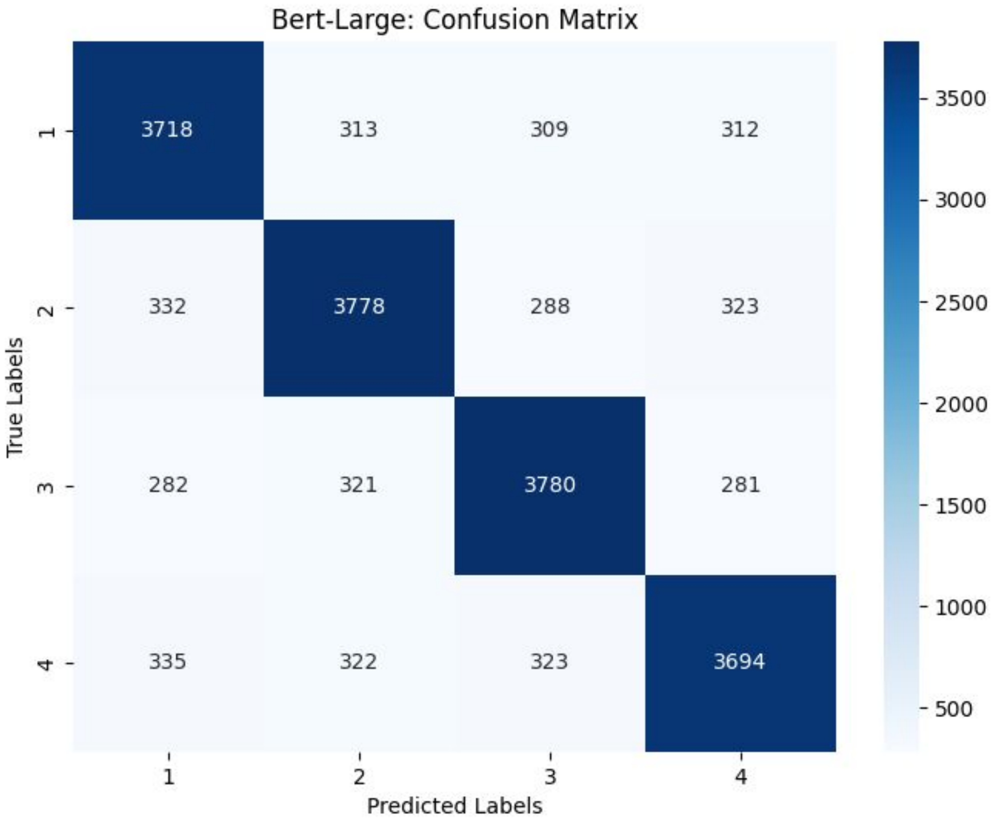
SWAG dataset

- **S**ituations **W**ith **A**dversarial **G**enerations.
- designed for grounded common sense inference
- Predict next plausible sequence/situation
  - Generated from video captions
- Multiple-choice format (1 out of 4 is correct)
  - AutoModelForMultipleChoice, own class for ModernBERT and GPT-2
  - Dropout, Linear Classifier (hidden dimension → 4) and Soft max
- Adversarial Filtering

SWAG results

Model	Validation		Test		Parameters
	Paper	Our	Paper	Our	
BERT-Base	81.6	76.6	-	76.1	110M
BERT-Large	86.6	79.3	86.3	80.8	340M
OpenAI GPT	-	66.8	78.0	66.2	117M
ModernBERT-base	-	77.6	-	77.8	149M
ModernBERT-base$_{xavier}$	-	77.7	-	78.0	149M
ModernBERT-large	-	82.0	-	82.1	395M
DeBERTa-v3-base	-	83.6	-	83.4	100M

SWAG



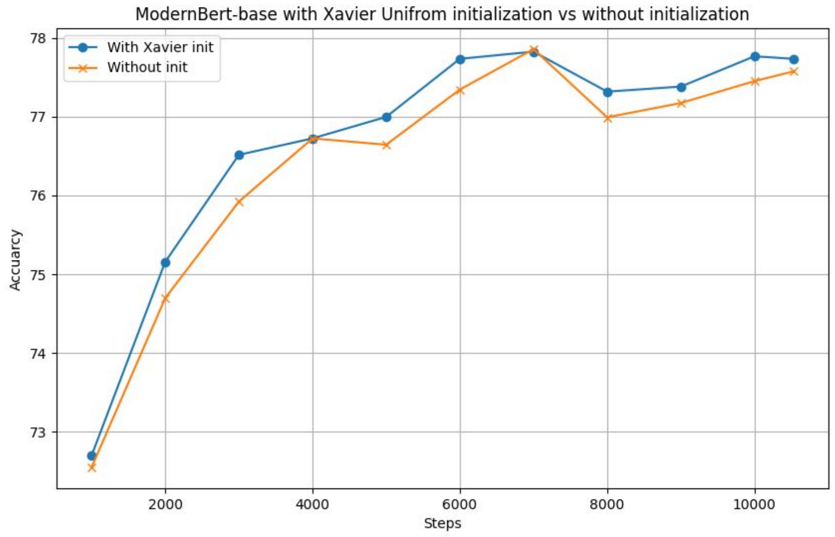
Model	ters
BERT-Bas	110M
BERT-Lar	340M
OpenAI G	117M
ModernBE	149M
ModernBE	149M
ModernBE	395M
DeBERTa-	100M

aining
plit,
e not

SWAG results

Model	Validation		Test		Parameters
	Paper	Our	Paper	Our	
BERT-Base	81.6	76.6	-	[illegible]	[illegible]
BERT-Large	86.6	79.3	86.3	[illegible]	340M
OpenAI GPT	-	66.8	78.0	66.2	117M
ModernBERT-base	-	77.6	-	77.8	149M
ModernBERT-base$_{xavier}$	-	77.7	-	78.0	149M
ModernBERT-large	-	82.0	-	82.1	395M
DeBERTa-v3-base	-	83.6	-	83.4	100M

⚠️ We used a different and smaller version of GPT



			Test		Parameters
			Paper	Our	
			-	76.1	110M
			86.3	80.8	340M
			78.0	66.2	117M
ModernBERT-base	-	77.6	-	77.8	149M
ModernBERT-base$_{xavier}$	-	77.7	-	78.0	149M
ModernBERT-large	-	82	-	82.1	395M
DeBERTa-v3-base	-			3.4	100M

🎉 We get slightly better results with Xavier-weight-initialization

SWAG results

Model	Validation		Test		Parameters
	Paper	Our	Paper	Our	
BERT-Base	81.6	76.6	-	76.1	110M
BERT-Large	86.6	79.3	86.3	80.8	340M
OpenAI GPT	-	66.8	78.0	66.2	117M
[illegible]	[illegible]	[illegible]	[illegible]	77.8	149M
[illegible]	[illegible]	[illegible]	[illegible]	78.0	149M
ModernBE[illegible]arge	-	82.0	-	82.1	395M
DeBERTa-v3-base	-	83.6	-	83.4	100M

🎉 Current SOTA: DeBERTa-Large with **94%**: Uses **disentangled attention** and an **enhanced mask decoder**

SNLI Fine-tuning

Stanford Natural Language Inference

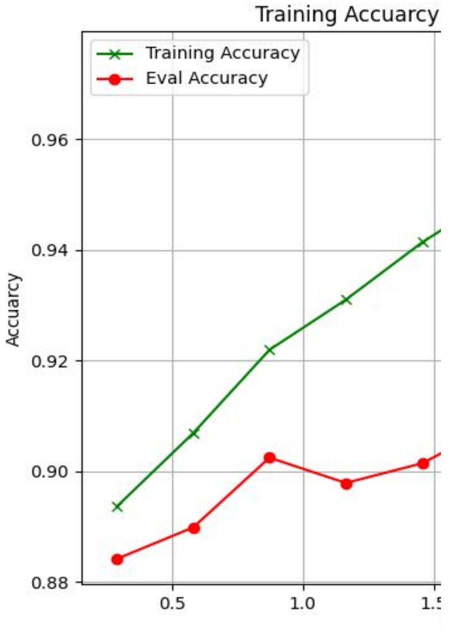
SNLI dataset

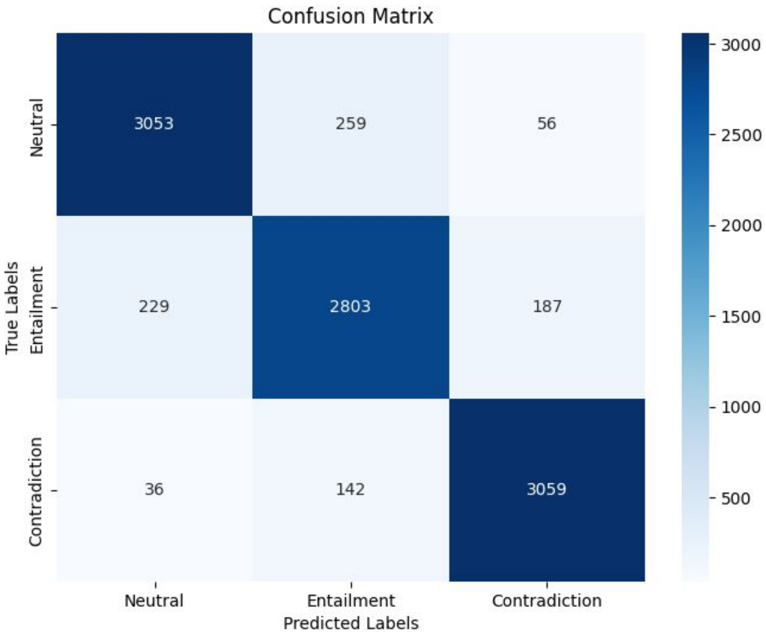
- **S**tanford **N**atural **L**anguage **I**nference
- 570k human written English sentence pairs
- Text + Hypothesis $\rightarrow$ Judgement (Contradiction, Neutral, Entailment)
- Custom Class $NLIClassifier$
  - Adds a linear classification layer after BERT
  - Would have been also possible with $AutoModelForSequenceClassification$

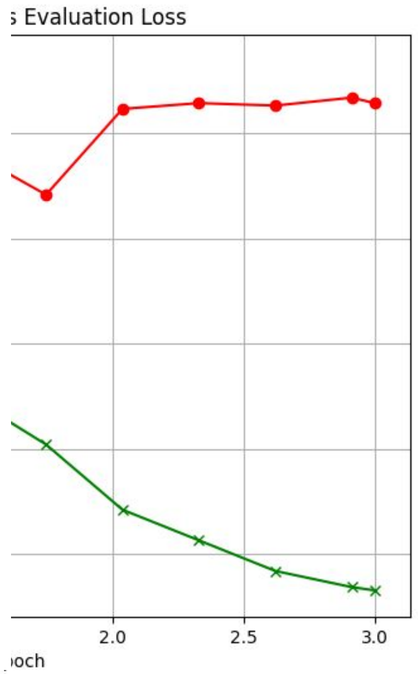
SNLI Results

» Very high accuracy for $\text{BERT}_{\text{BASE}}$: 90.4%

⚠️ Due to the size (570k) and high performance of $\text{BERT}_{\text{BASE}}$, we didn't finetune $\text{BERT}_{\text{LARGE}}$





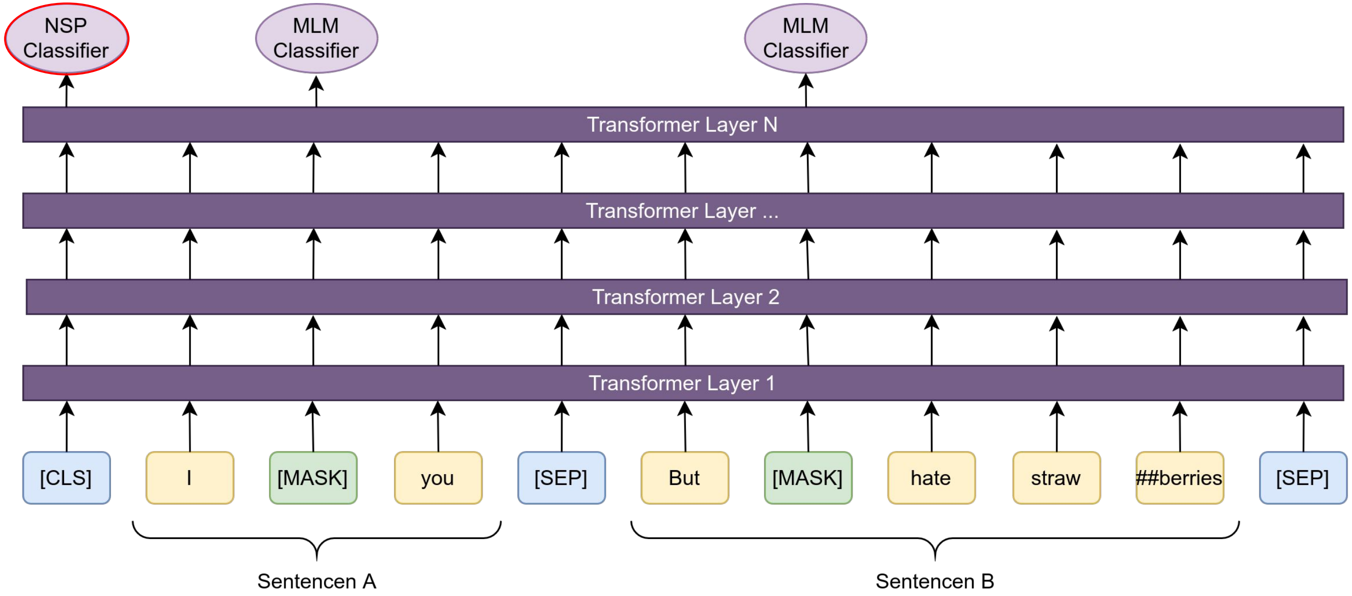


Thank you for your attention.

Implementation of experiments:
https://github.com/markhun/2024W-DLNLP-BERT

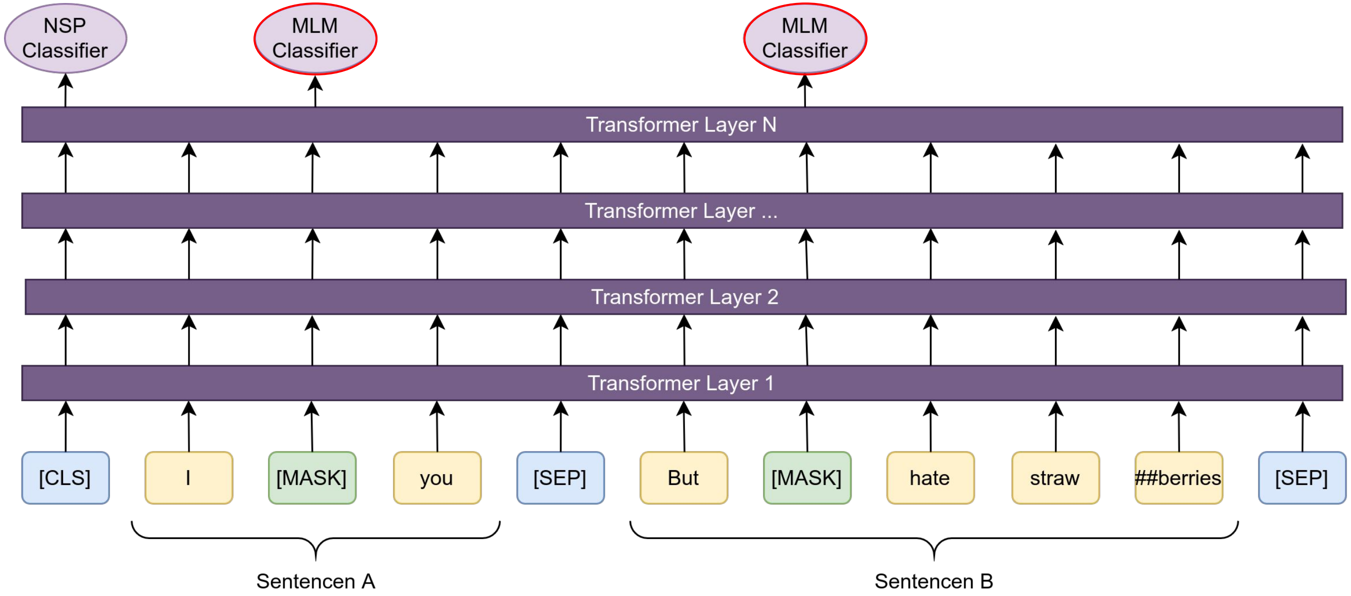
Pre-Training

- ... send it through the model and
- ... give **[CLS]** token result to Next-Sentence-Prediction (NSP) classifier



Pre-Training

- … send it through the model and
- … give **[MASK]** token results to Masked-Language-Modeling (MLM) classifier



GLUE Fine-tuning (extended)

General **L**anguage **U**nderstanding **E**valuation

GLUE[2]

General **L**anguage **U**nderstanding **E**valuation

» A whole suite of 11 sentence classification problems in English.

- 1 Diagnostic Set (**AX**)
- 7 classification problems with 2 labels (**CoLA**, **MRPC**, **QNLI**, **QQP**, **RTE**, **SST2**, **WNLI**)
- 2 classification problems with 3 labels (**MNLI-m**, **MNLI-mm**)
- 1 regression problem within values of 0 to 5 (**STSB**)

» A set of standardized public datasets.

- 9 training datasets (Problems **MNLI-m**, **MNLI-mm** and **AX** share the same training dataset)
- 10 validation datasets (No validation dataset provided for **AX**)
- 11 tests sets with censored labels

[2] Wang, Alex, et al. "GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding." 2018.

GLUE[2]

General **L**anguage **U**nderstanding **E**valuation

» A whole suite of 11 sentence classification problems in English.

⚠️ Considered problematic as it includes adversarial items

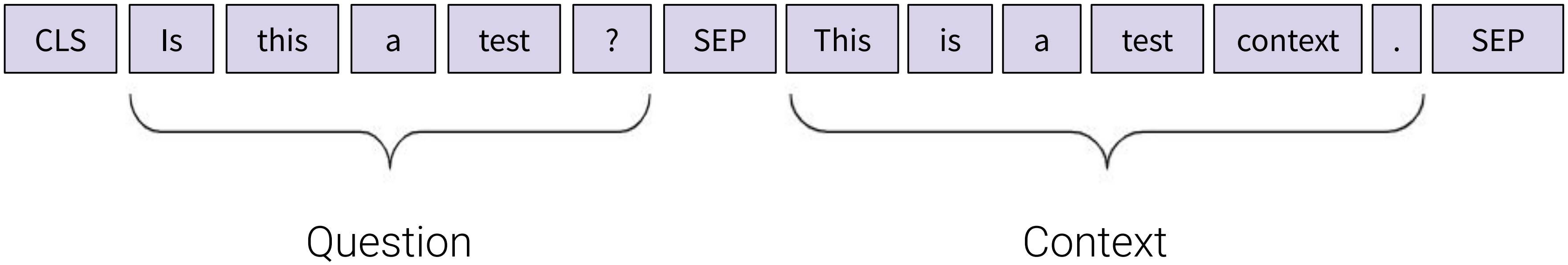
- ~~1 Diagnostic Set (**AX**)~~
- 6 ~~7~~ classification problems with 2 labels (**CoLA**, **MRPC**, **QNLI**, **QQP**, **RTE**, **SST2**, ~~**WNLI**~~)
- 2 classification problems with 3 labels (**MNLI-m**, **MNLI-mm**)
- 1 regression problem within values of 0 to 5 (**STSB**)

» A set of standardized public datasets.

- 9 training datasets (Problems **MNLI-m**, **MNLI-mm** and **AX** share the same training dataset)
- 10 validation datasets (No validation dataset provided for **AX**)
- 11 tests sets with censored labels

[2] Wang, Alex, et al. "GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding." 2018.

Input sequence for SQuAD 1.1



Example of the training dataset

```
{
    "answers": {
        "answer_start": [1],
        "text": ["This is a test text"]
    },
    "context": "This is a test context.",
    "id": "1",
    "question": "Is this a test?",
    "title": "train test"
}
```

- Validation dataset allows several possible answers for each sample